



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/533,735	05/04/2005	Robert Jochemsen	NL 021103	4556
65913	7550	08/07/2008	EXAMINER	
NXP, B.V. NXP INTELLECTUAL PROPERTY DEPARTMENT M/S41-SJ 1109 MCKAY DRIVE SAN JOSE, CA 95131			SCHNEE, HAL W	
			ART UNIT	PAPER NUMBER
			2186	
			NOTIFICATION DATE	DELIVERY MODE
			08/07/2008	ELECTRONIC

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ip.department.us@nxp.com

### Office Action Summary

**Application No.**

10/533,735

**Applicant(s)**

JOICHEMSEN ET AL.

**Examiner**

HAL SCHNEE

**Art Unit**

2186

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 09 July 2008.  
2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.  
3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-8 and 10-25 is/are pending in the application.  
4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.  
5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.  
6) ☒ Claim(s) 1-8 and 10-25 is/are rejected.  
7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.  
8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.  
10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).  
11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).  
a) ☐ All b) ☐ Some \* c) ☐ None of:  
1. ☐ Certified copies of the priority documents have been received.  
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.  
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- 1) ☐ Notice of References Cited (PTO-892)  
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)  
3) ☐ Information Disclosure Statement(s) (PTO-8508)  
Paper No(s)/Mail Date \_\_\_\_\_  
4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date \_\_\_\_\_  
5) ☐ Notice of Informal Patent Application  
6) ☐ Other: \_\_\_\_\_

### DETAILED ACTION

1. Claims 1-8 and 10-25 are pending in this application.
2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

3. Claims 1-4, 7, 8, and 10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Stockdale et al. (U.S. Patent 6,804,763, hereafter “Stockdale”) in view of Cheng et al. (U.S. Patent 5,701,516, hereafter “Cheng”), and further in view of O’Neill (U.S. 2003/0182414).

**Regarding Claim 1**, Stockdale teaches a memory management device for managing a memory space of at least one persistent-memory device (fig. 4; col. 4, lines 15-17—“non-volatile memory storage device”), comprising a memory allocation unit (col. 10, lines 52-55—“NV-RAM manager”) adapted to communicate with at least one application device and to allocate at least one first part of said memory space to said application device (col. 4, lines 17-25; application devices are shown in fig. 2, item 215), wherein said allocation unit is further adapted to communicate with at least one file system device, and to allocate on request from said application device or from said file system device said first part of said memory space to said file system (col. 7, lines 19-26).

Stockdale does not teach that the persistent memory is used as a write cache memory for said file system. However, Cheng teaches a persistent memory used as a write cache memory for said file system (col. 2, lines 57-61).

All of these claimed elements were known in Stockdale and Cheng and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the persistent memory used as write cache of Cheng with the device of Stockdale to yield the predictable result of a memory management device in which the persistent memory is used as a write cache for a file system device. One would be motivated to make this combination for the purpose of increasing the speed of writes by writing data to be stored to the non-volatile memory rather than a slower storage device.

Stockdale/Cheng does not specifically teach using the allocated memory space to write a first working data structure comprising a plurality of working data blocks to the memory space and to write a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure.

However, O'Neill teaches using the allocated memory space to write a first working data structure comprising a plurality of working data blocks to the memory space and to write a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure (fig. 10, memory blocks 1234 and 1242; ¶ [0148]—first working data structure 1234 is written to persistent {flash} memory 1002. An additional copy is made in the form of a second working data structure to location 1242 of persistent memory 1002 to provide additional fault tolerance. Since 1234 and 1242 are contained in the same persistent memory 1002, they can be considered as residing in the same memory

space; since they are identical copies of the same data, the second working data structure comprises a copy of the working data blocks of the first working data structure).

All of the claimed elements were thus known in Stockdale/Cheng and O'Neill and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the working data structures of O'Neill with the persistent memory allocation of Stockdale/Cheng to yield the predictable result of a memory management device that allocates a part of the persistent memory to write a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure. One would be motivated to make this combination for the purpose of securing a copy of the working data while continuing to work on the original working data.

**Regarding Claim 2**, Stockdale teaches said memory allocation unit is adapted to maintain a memory allocation table at a current status, said memory allocation table assigning at least one memory address representing a defined part of said memory space to either said application device or to said file system device (fig. 9; col. 29, lines 17-19).

**Regarding Claim 3**, Stockdale teaches a memory management device according to claim 2, further comprising a processor and a memory (fig. 3, processor 300 and memories 325 and 355), wherein said memory allocation unit is implemented in the form of at least one first executable file contained in said memory (col. 10, lines 52-55—"NV-RAM Manager").

**Regarding Claim 4**, Stockdale teaches said memory is a persistent-memory device, in particular said persistent-memory device (col. 33, lines 11-14—software is installed in the NV-RAM and executed on the processor, as shown in fig. 11, item 1140).

**Regarding Claim 7**, Stockdale teaches an application device, comprising a persistent-memory device connected to a processor (fig. 3, processor 300 and persistent-memory device 355; col. 17, line 64-col. 18, line 3), and a data management unit adapted to manipulate data in said persistent memory device (fig. 4; col. 4, lines 15-17—“non-volatile memory storage device”), wherein said data management unit is adapted to write at least one third executable file to said persistent memory device, or to provide the file system with a reference to at least on third executable file in said file system (col. 17, lines 38-41 shows adding an executable file to the persistent memory device; col. 35, lines 21-34 shows that the device can add any number of executable files), such that by executing said third executable file said processor is adapted to transform said data into a predetermined data-sequence form (col. 7, lines 45-49 and col. 35, lines 14-20—the processor can execute a compression utility, which transforms data into a predetermined data-sequence form; all executable files used by the processor are stored in the persistent memory device).

Stockdale does not teach that the persistent memory is used as a write cache memory for said file system. However, Cheng teaches a persistent memory used as a write cache memory for said file system (col. 2, lines 57-61).

All of these claimed elements were known in Stockdale and Cheng and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the

persistent memory used as write cache of Cheng with the device of Stockdale to yield the predictable result of an application device in which the persistent memory is used as a write cache for a file system device. One would be motivated to make this combination for the purpose of increasing the speed of writes by writing data to be stored to the non-volatile memory rather than a slower storage device.

Stockdale/Cheng does not teach transforming a copy of a first working data structure into a predetermined data-sequence form, wherein the copy of the first working data structure is stored in a same memory space as the first working data structure. However, O'Neill teaches making a copy of the first working data structure is stored in a same memory space as the first working data structure (fig. 10, memory blocks 1234 and 1242; ¶ [0148]—first working data structure 1234 is written to persistent {flash} memory 1002. An additional copy is made in the form of a second working data structure to location 1242 of persistent memory 1002 to provide additional fault tolerance. Since 1234 and 1242 are contained in the same persistent memory 1002, they can be considered as residing in the same memory space; since they are identical copies of the same data, the second working data structure comprises a copy of the working data blocks of the first working data structure).

All of the claimed elements were thus known in Stockdale/Cheng and O'Neill and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the working data structures of O'Neill with the data form transformation of Stockdale/Cheng to yield the predictable result of an application device in which the processor is adapted to transform a copy of a first working data structure into a predetermined data-sequence

form, wherein the copy of the first working data structure is stored in a same memory space as the first working data structure. One would be motivated to make this combination for the purpose of securing a copy of the working data while continuing to work on the original working data.

**Regarding Claim 8**, Stockdale teaches said data management unit is provided in the form of least one fourth executable file in a memory, particularly, in said persistent memory (col. 10, lines 52-55—“NV-RAM Manager” and col. 33, lines 11-14—software is installed in the NV-RAM and executed on the processor, as shown in fig. 11, item 1140).

**Regarding Claim 10**, Stockdale teaches a data processing system (fig. 3), comprising a memory management device for managing a memory space of at least one persistent-memory device (fig. 4; col. 4, lines 15-17—“non-volatile memory storage device”), comprising a memory allocation unit (col. 10, lines 52-55—“NV-RAM manager”) adapted to communicate with at least one application device and to allocate at least one first part of said memory space to said application device (col. 4, lines 17-25; application devices are shown in fig. 2, item 215), wherein said allocation unit is further adapted to communicate with at least one file system device, and to allocate on request from said application device or from said file system device said first part of said memory space to said file system (col. 7, lines 19-26).

Stockdale does not teach that the persistent memory is used as a write cache memory for said file system. However, Cheng teaches a persistent memory used as a write cache memory for said file system (col. 2, lines 57-61).

All of the claimed elements were known in Stockdale and Cheng and could have been combined by known methods with no change in their respective functions. It therefore would



have been obvious to a person of ordinary skill in the art at the time of invention to combine the persistent memory used as write cache of Cheng with the system of Stockdale to yield the predictable result of a data processing system in which the persistent memory is used as a write cache for a file system device. One would be motivated to make this combination for the purpose of increasing the speed of writes by writing data to be stored to the non-volatile memory rather than a slower storage device.

Stockdale/Cheng does not specifically teach using the allocated memory space to write a first working data structure comprising a plurality of working data blocks to the memory space and to write a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure.

However, O'Neill teaches using the allocated memory space to write a first working data structure comprising a plurality of working data blocks to the memory space and to write a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure (fig. 10, memory blocks 1234 and 1242; ¶ [0148]—first working data structure 1234 is written to persistent {flash} memory 1002. An additional copy is made in the form of a second working data structure to location 1242 of persistent memory 1002 to provide additional fault tolerance. Since 1234 and 1242 are contained in the same persistent memory 1002, they can be considered as residing in the same memory space; since they are identical copies of the same data, the second working data structure comprises a copy of the working data blocks of the first working data structure).

All of the claimed elements were thus known in Stockdale/Cheng and O'Neill and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the working data structures of O'Neill with the persistent memory allocation of Stockdale/Cheng to yield the predictable result of a data processing system that allocates a part of the persistent memory to write a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure. One would be motivated to make this combination for the purpose of securing a copy of the working data while continuing to work on the original working data.

4. Claims 5, 6, and 11-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Stockdale (U.S. Patent 6,804,763) in view of Cheng (U.S. Patent 5,701,516), further in view of O'Neill (U.S. 2003/0182414) and further in view of Hanes (U.S. 2003/0081932).

**Regarding Claim 5**, Stockdale teaches a file system device adapted to maintain a file allocation table at a current status, said file allocation table assigning at least one disk space address to at least one file, wherein said file allocation Unit is adapted to communicate with a memory management device that is related to a persistent-memory device and to include an address of at least one first memory space of said persistent-memory device in the maintenance of said file allocation table (col. 34, lines 51-58—the same techniques are used for the file system device as for memory management for the application device. Fig. 12 and col. 35, lines 14-20 show the files; fig. 9 shows the file allocation table).

In the device of Stockdale, the functions of the file system device are performed by the same device/processor as the functions of the application device; Stockdale thus does not specifically teach a separate file system device, comprising a file allocation unit; or that the persistent memory is used as a write cache memory for said file system device. However, Cheng teaches a file system device, comprising a file allocation unit (fig. 2, Storage Processor 38 {shown in detail in fig. 3}; col. 5, lines 34-37); and the persistent memory is used as a write cache memory for said file system device (col. 2, lines 57-61).

All of these claimed elements were known in Stockdale and Cheng and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the file system device comprising a file allocation unit and the persistent memory used as write cache of Cheng with the device of Stockdale to yield the predictable result of a file system device with a file allocation unit, in which the persistent memory is used as a write cache for the file system device. One would be motivated to make this combination for the purpose of increasing the speed of writes by writing data to be stored to the non-volatile memory rather than a slower storage device, and to free the application device to perform other tasks by offloading file management to a file allocation unit.

Stockdale/Cheng does not teach wherein the file system device is configured to convert a copy of a first working data structure to a file data structure and to write the file data structure to a secondary storage medium, wherein the copy of the first working data structure is written to a same memory space as the first working data structure.

However, O'Neill teaches using the allocated memory space to write a first working data structure comprising a plurality of working data blocks to the memory space and to write a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure (fig. 10, memory blocks 1234 and 1242; ¶ [0148]—first working data structure 1234 is written to persistent {flash} memory 1002. An additional copy is made in the form of a second working data structure to location 1242 of persistent memory 1002 to provide additional fault tolerance. Since 1234 and 1242 are contained in the same persistent memory 1002, they can be considered as residing in the same memory space; since they are identical copies of the same data, the second working data structure comprises a copy of the working data blocks of the first working data structure).

All of these claimed elements were thus known in Stockdale/Cheng and O'Neill and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the working data structures of O'Neill with the persistent memory allocation of Stockdale/Cheng to yield the predictable result of a file system device that allocates a part of the persistent memory to write a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure. One would be motivated to make this combination for the purpose of securing a copy of the working data while continuing to work on the original working data.

Stockdale/Cheng/O'Neill does not teach the file system device is configured to convert a copy of a first working data structure to a file data structure and to write the file data structure to a secondary storage medium. However, Hanes teaches the file system device is configured to convert a working data structure to a file data structure and to write the file data structure to a secondary storage medium (fig. 2A; ¶ [0017]—CD Recorder Engine 20 {file system device} converts working data from application formats to a file system format for storage on a CD. An embodiment of the interface between the application device and the file system device is described in ¶ [0014], and recording to CD {writing the file structure to a secondary storage medium} is described in ¶ [0019]).

All of the claimed elements were thus known in Stockdale/Cheng/O'Neill and Hanes and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the conversion to a file system format and writing to CD of Hanes with the working data copy of Stockdale/Cheng/O'Neill to yield the predictable result of a file system device that is configured to convert a copy of a first working data structure to a file data structure and to write the file data structure to a secondary storage medium. One would be motivated to make this combination for the purpose of securing the working data on a secondary medium for long-term storage.

**Regarding Claim 6**, Stockdale teaches a file system device according to claim 5, further comprising a processor and a memory (fig. 3, processor 300 and memories 325 and 355), wherein said memory allocation unit is implemented in the form of at least one second executable file contained in said memory (col. 10, lines 52-55—"NV-RAM Manager").

**Regarding Claim 11**, Stockdale teaches a method for managing memory space of a persistent-memory device (col. 6, lines 5-6), comprising allocating at least one first part of said memory space to a file system device (col. 6, lines 11-13 and col. 7, lines 23-26); and

converting a working data structure into a predetermined data sequence form (col. 7, lines 45-49 and col. 35, lines 14-20—the processor can execute a compression utility, which transforms data into a predetermined data-sequence form).

Stockdale does not teach that the persistent memory is used as a write cache memory for said file system. However, Cheng teaches a persistent memory used as a write cache memory for said file system (col. 2, lines 57-61).

All of these claimed elements were known in Stockdale and Cheng and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the persistent memory used as write cache of Cheng with the system of Stockdale to yield the predictable result of a data processing system in which the persistent memory is used as a write cache for a file system device. One would be motivated to make this combination for the purpose of increasing the speed of writes by writing data to be stored to the non-volatile memory rather than a slower storage device.

Stockdale/Cheng does not teach:

writing a first working data structure comprising a plurality of working data blocks to the memory space;

writing a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure;

and does not specifically teach converting the second working data structure into a predetermined data sequence form.

However, O'Neill teaches:

writing a first working data structure comprising a plurality of working data blocks to the memory space; and writing a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure (fig. 10, memory blocks 1234 and 1242; ¶ [0148]—first working data structure 1234 is written to persistent {flash} memory 1002. An additional copy is made in the form of a second working data structure to location 1242 of persistent memory 1002 to provide additional fault tolerance. Since 1234 and 1242 are contained in the same persistent memory 1002, they can be considered as residing in the same memory space; since they are identical copies of the same data, the second working data structure comprises a copy of the working data blocks of the first working data structure).

All of the claimed elements were thus known in Stockdale/Cheng and O'Neill and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the working data structures of O'Neill with the method of Stockdale/Cheng to yield the predictable result of a method for managing memory space of a persistent memory device that

allocates a part of the persistent memory to write a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure, and converts the second working data structure into a predetermined data sequence form. One would be motivated to make this combination for the purpose of securing a copy of the working data while continuing to work on the original working data.

**Regarding Claim 12**, Stockdale teaches said allocating step comprises a step of blocking a writing access to said first part of said memory space (col. 6, lines 54-61 and col. 32, lines 22-24—restricting access means blocking writing access for certain devices or processes).

**Regarding Claim 13**, Stockdale teaches said allocating step comprises a step of giving away to said file system device the power of reading access to said first part of said memory space (col. 6, lines 46-50—sending a handle constitutes giving away reading access since the handle is required to access the allocated memory area).

**Regarding Claim 14**, Stockdale teaches a step of deallocating said first part of said memory space to a memory management device (col. 6, lines 41-43 and 54-58—the method includes deallocating among the functions).

**Regarding Claim 15**, Stockdale teaches said allocating step or said deallocating step comprises transmitting an address range defining said first part of said memory space from said memory management device to said file system device or, respectively, vice versa (col. 28, lines 18-20 and col. 29, lines 22-24—the method transmits a handle to a memory block of predefined size; this defines the range of addresses that is transmitted).



5. Claims 16-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Stockdale (U.S. Patent 6,804,763) in view of Cheng (U.S. Patent 5,701,516) in view of O'Neill (U.S. 2003/0182414) in view of Hanes (U.S. 2003/0081932), as applied to Claim 14, above, and further in view of Lee et al. (U.S. Patent 5,930,167, hereafter "Lee").

**Regarding Claim 16**, Stockdale/Cheng/O'Neill/Hanes teaches said deallocating step is performed for said first part of said memory space based on any number of conditions (Stockdale, col. 32, lines 60-65—various flags govern when deallocation is and is not allowed).

Stockdale/Cheng/O'Neill/Hanes does not specifically teach that deallocation is performed given the condition that first data contained in said first part of said memory space is stored in the form of file data in a second part of said memory space, said file data having a predetermined file structure, and that said second part of said memory space is allocated to said file system device.

However, Lee teaches that first data contained in said first part of said memory space is stored in the form of file data in a second part of said memory space, said file data having a predetermined file structure, and that said second part of said memory space is allocated to said file system device (col. 5, line 60-col. 6, line 6; figs. 4A-4D).

All of the claimed elements were known in Stockdale/Cheng/O'Neill/Hanes and Lee, and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the storing data of Lee with the conditional deallocation of Stockdale/Cheng/O'Neill/Hanes to yield the predictable result of performing the deallocating step given the condition that first data contained in said first part of said memory space is stored

in the form of file data in a second part of said memory space, said file data having a predetermined file structure, and that said second part of said memory space is allocated to said file system device. One would be motivated to make this combination for the purpose of securing a copy of the working data before deallocating the memory used to store the working data.

**Regarding Claim 17**, Stockdale/Cheng/O'Neill/Hanes teaches said deallocating step is performed for said second part of said memory space given the condition that said file data has been written to a secondary storage medium (Cheng, col. 6, lines 42-48—the flush operation writes the data to a disk, a secondary storage medium).

**Regarding Claim 18**, Stockdale/Cheng/O'Neill/Hanes teaches the step of Claim 17, as described above, but does not teach a method for write-caching first data worked on by an application, said first data being contained in a first part of a memory space of a persistent-memory device. However, Lee teaches a method for write-caching first data worked on by an application, said first data being contained in a first part of a memory space of a persistent-memory device (Lee, Abstract, lines 1-6 and col. 8, lines 10-14).

All of the claimed elements were known in Stockdale/Cheng/O'Neill/Hanes and Lee, and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the method for write caching of Lee with the method of Stockdale/Cheng to yield the predictable result of performing a memory managing method according to claim 17 comprised in a method for write-caching first data worked on by an application, said first data being contained in a first part of a memory space of a persistent-memory device.

**Regarding Claim 19**, Stockdale/Cheng/O'Neill/Hanes teaches after said allocating step, a step of sending a confirmation message from said file system device to said application device (Cheng, col. 7, lines 50-59—after performing an operation, a confirmation message is sent to the application device {host processor}; and col. 13, lines 51-55—enabling the NVRAM constitutes an allocating step).

**Regarding Claim 20**, Stockdale/Cheng/O'Neill/Hanes teaches flags determining when deallocation is allowed (Stockdale, col. 32, lines 60-65) and additional data stored in the persistent memory (col. 14, lines 44-50 and 62-63; this constitutes the third data of the present claim), but Stockdale/Cheng/O'Neill/Hanes does not teach said first data is a copy of third data contained in a third part of said memory space, said write-caching method comprising, before performing said memory managing method, a step of copying said third data to said first memory space.

However, Lee teaches said first data is a copy of third data contained in a third part of said memory space, said write-caching method comprising a step of copying said third data to said first memory space (col. 5, line 60-col. 6, line 6; figs. 4A-4D shows the copying operation before the memory management method {i.e. before deallocating the memory}).

All of the claimed elements were known in Stockdale, Cheng, and Lee, and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the data copying and write caching method of Lee with the determination of when deallocation can be performed of Stockdale/Cheng to yield the predictable result of a write

caching method that copies third data to the first memory space before performing a memory management method.

**Regarding Claim 21**, Stockdale/Cheng/O'Neill/Hanes teaches:

allocating a fourth part of said memory space to said application device for an executable file or dynamic link library (Stockdale, col. 17, lines 38-41 shows adding an executable file to the persistent memory device; col. 35, lines 21-34 shows that the device can add any number of executable files);

writing said executable file or dynamic link library to said fourth part of said memory space (Stockdale, col. 33, lines 11-14—any number of executables can be stored in any number of parts of the memory space); and

allocating said fourth part of said memory space to said file system device (Stockdale, col. 7, lines 19-26 and col. 34, lines 51-58—any number of parts of the memory space can be allocated to the file system device).

Stockdale/Cheng/O'Neill/Hanes does not teach that the executable file is adapted to converting said first data into file data. However, Lee teaches the executable file is adapted to converting said first data into file data (col. 5, line 60-col. 6, line 6 and figs. 4A-4D show an executable adapted to converting data into file data).

All of the claimed elements were known in Stockdale/Cheng/O'Neill/Hanes and Lee and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the executable file of Lee with the memory allocation and writing of

Stockdale/Cheng to yield the predictable result of method that allocates memory to the file system device and writes an executable file that is adapted to converting data into file data.

**Regarding Claim 22**, Stockdale/Cheng/O'Neill/Hanes does not teach a step of transforming said first data into said file data with the aid of said executable file or said dynamic link library. However, Lee teaches a step of transforming said first data into said file data with the aid of said executable file or said dynamic link library (col. 5, line 60-col. 6, line 6).

All of the claimed elements were known in Stockdale/Cheng/O'Neill/Hanes and Lee and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the transforming data of Lee with the method of Stockdale/Cheng/O'Neill/Hanes to yield the predictable result of method that includes a step of transforming said first data into said file data with the aid of said executable file or said dynamic link library.

**Regarding Claim 23**, Stockdale/Cheng/O'Neill/Hanes does not teach said transforming step is initiated by said file system device. However, Lee teaches said transforming step is initiated by said file system device (col. 6, lines 12-19—performing the transforming step when the memory system is idle is necessarily initiated by the file system device since the application device has no way of knowing when the memory system is idle).

All of the claimed elements were known in Stockdale/Cheng/O'Neill/Hanes and Lee and could have been combined by known methods with no change in their respective functions. It therefore would have been obvious to a person of ordinary skill in the art at the time of invention to combine the initiation of Lee with the method of Stockdale/Cheng to yield the predictable result of method in which the transforming step is initiated by the file system device.

**Regarding Claim 24**, Stockdale/Cheng/O'Neill/Hanes teaches after said transforming step, a step of deallocating said fourth part of said memory space to said memory management device (Stockdale, col. 6, lines 41-43 and 54-58—the method includes deallocating among the functions, and it can be performed after any step in the process).

**Regarding Claim 25**, Stockdale/Cheng/O'Neill/Hanes teaches a method for saving data worked on by an application device to a file on a secondary storage medium (Cheng, col. 2, lines 57-61—data storage devices are a secondary storage medium), comprising writing said file data to said secondary storage medium (Cheng, col. 7, lines 25-29).

#### ***Response to Arguments***

6. Applicant's arguments filed 9 July 2008 have been fully considered but they are not persuasive.

**Regarding Claim 1**, the applicant argues that O'Neill (U.S. 2003/0182414) does not teach using the allocated memory space to write a first working data structure comprising a plurality of working data blocks to the memory space and to write a second working data structure comprising a copy of the plurality of working data blocks, wherein the second working data structure comprises a copy of the first working data structure in the same memory space as the first working data structure, as recited by the present claim, asserting that O'Neill's copying of data into banks 1234 and 1242 of persistent memory device 1002 (described it ¶ [0148] of O'Neill) does not teach the limitations of the present claim. The examiner agrees with the applicant's argument that the copying of ¶ [0140] and [0142] of O'Neill is indeed from one memory device (RAM 1004) to another (flash {or persistent} memory 1002). However, the

examiner maintains that O'Neill does teach the claim limitations in question and has clarified the citations of O'Neill in the rejections above, which now explain that memory blocks 1234 and 1242 in fig. 10 are both copies of the same working data structure—they are identical copies of the working data structure from RAM 1004, and are contained in the same memory space.

It is the examiner's view that the arguments presented center on details that may be the applicant's intention, but are not contained in the present claim. The applicant argues that O'Neill does not copy data from one location in the persistent memory device to another location in the persistent memory device. However, the present claim does not require copying in this manner; it only specifies that the second working data structure be a copy of the data contained in the first working data structure. As the applicant states, O'Neill copies data from RAM 1004 to both locations 1234 and 1242 in persistent memory 1004 (O'Neill, ¶ [0148]). The result is that both locations 1234 and 1242 will contain identical copies of the working data structure; the second working data structure is therefore a copy of the first, as required by the present claim.

The applicant also argues that locations 1234 and 1242 are contained in separate memory devices. However, the present claim makes no mention of memory devices, only a memory space; the two are not necessarily synonymous. A DIMM, for example, may contain several memory devices, but the DIMM (or even several DIMMs) can be part of the same memory space. Paragraph [0030] of the applicant's disclosure explains that a memory space may comprise a plurality of memory devices. Locations 1234 and 1242 of O'Neill can therefore be considered as part of the same memory space, referred to as memory space 1002.

**Regarding independent Claims 5, 7, 10, and 11,** the examiner maintains that O'Neill teaches the limitations similar to those of Claim 1 in the same manner as for Claim 1.

**Regarding the dependent claims**, the rejections are maintained as the rejections of the independent claims are maintained.

***Conclusion***

7. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to HAL SCHNEE whose telephone number is (571)270-1918. The examiner can normally be reached on Monday-Friday 8:00 a.m. to 4:30 p.m. E.S.T..

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matthew M. Kim can be reached on (571) 272-4182. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.



Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Matt Kim/  
Supervisory Patent Examiner, Art Unit  
2186

HWS 28 July 2008